Package ‘GMDHreg’

July 26, 2019

Type Package
Title Regression using GMDH Algorithms
Version 0.2.0
Author Manuel Villacorta Tilve
Maintainer Manuel Villacorta Tilve <mvt.oviedo@gmail.com>
License GPL-3
Encoding UTF-8
LazyData true
Depends R (>= 2.15)
Imports stats, utils
Suggests knitr, rmarkdown
VignetteBuilder knitr
RoxygenNote 6.1.1
NeedsCompilation no
Repository CRAN
Date/Publication 2019-07-26 08:10:04 UTC
**R topics documented:**

- gmdh.combi ................................................. 2
- gmdh.combi.twice ........................................ 3
- gmdh.gia .................................................. 5
- gmdh.mia .................................................. 7
- predict.combi ............................................. 9
- predict.combitwice ...................................... 10
- predict.gia .............................................. 11
- predict.mia .............................................. 12

---

**gmdh.combi**

---

**GMDH Combinatorial**

---

**Description**

Build a regression model performing GMDH Combinatorial.

This is the basic GMDH algorithm. For more information, please read the package’s vignette.

**Usage**

```r
gmdh.combi(X, y, G = 2, criteria = c("PRESS", "test", "ICOMP"),
            x.test = NULL, y.test = NULL)
```

**Arguments**

- **X**
  - matrix with N>1 columns and M rows, containing independent variables in the model.
  - Be careful, N>4 and G=2, could be computationally very expensive and time consuming.
  - The data must not contain NAs

- **y**
  - vector or matrix containing dependent variable in the model.
  - The data must not contain NAs

- **G**
  - polynomial degree.
  - 0: linear regression without quadratic and interaction terms.
  - 1: linear regression with interaction terms.
  - 2: original Ivakhnenko quadratic polynomial

- **criteria**
  - GMDH external criteria. Values:
    - PRESS: Predicted Residual Error Sum of Squares. It take into account all information in data sample and it is computed without recalculating of system for each test point.
    - test: use x.test and y.test to estimate RMSE (Root Mean Square Errors).
    - ICOMP: Index of Informational Complexity. Like PRESS, it is computed without recalculating of system.
x.test matrix with a sample randomly drawn from the initial data. This sample should not be included in X. It is used when criteria = test.

y.test vector or matrix with y values correspond with x.test values.

Value

An object of class 'combi'. This is a list with two elements: results and G. Results is a list with two elements:

- coef: coefficients of final selected GMDH Combinatorial model.
- CV: external criteria value for selected model.

G the grade of polynomial used in GMDH Combinatorial model.

References


Examples

```r
set.seed(123)
x <- matrix(data = c(rnorm(1050)), ncol = 3, nrow = 350)
colnames(x) <- c("a", "b", "c")
y <- matrix(data = c(10 + x[, "a"] + x[, "b"]^2 + x[, "c"]^3), ncol = 1)
colnames(y) <- "y"
x.test <- x[1:10, ]
y.test <- y[1:10]
x <- x[-c(1:10), ]
y <- y[-c(1:10)]

mod <- gmdh.combi(X = x, y = y, criteria = "PRESS")
pred <- predict(mod, x.test)
summary(sqrt((pred - y.test)^2))
```
gmdh.combi.twice  GMDH Twice-Multilayered Combinatorial

Description

Build a regression model performing GMDH Twice-Multilayered Combinatorial (TMC).
For more information, please read the package’s vignette.

Usage

gmdh.combi.twice(X, y, criteria = c("PRESS", "test", "ICOMP"), G = 2, x.test = NULL, y.test = NULL)

Arguments

X  
matrix with N>1 columns and M rows, containing independent variables in the
model.
Be careful, N>4 and G=2, could be computationally very expensive and time
consuming.
The data must not contain NAs

y  
vector or matrix containing dependent variable in the model.
The data must not contain NAs

criteria  
GMDH external criteria. Values:

  • PRESS: Predicted Residual Error Sum of Squares. It take into account all
information in data sample and it is computed without recalculating of sys-
tem for each test point.

  • test: use x.test and y.test to estimate RMSE (Root Mean Square Errors).

  • ICOMP: Index of Informational Complexity. Like PRESS, it is computed
without recalculating of system.

G  
polynomial degree.
0: linear regression without quadratic and interaction terms.
1: linear regression with interaction terms.
2: original Ivakhnenko quadratic polynomial.

x.test  
matrix with a sample randomly drawn from the initial data. This sample should
not be included in X.
It is used when criteria = test.

y.test  
vector or matrix with y values correspond with x.test values.
Value

An object of class `combitwice`. This is a list with two elements: results and G

Results is a list with two elements:

- coef: coefficients of final selected GMDH Combinatorial model.
- CV: external criteria value for selected model.

G the grade of polynomial used in GMDH Combinatorial model.

References


Examples

```r
set.seed(123)
x <- matrix(data = c(rnorm(1050)), ncol = 3, nrow = 350)
colnames(x) <- c("a", "b", "c")
y <- matrix(data = c(10 + x[, "a"] + x[, "b"]^2 + x[, "c"]^3), ncol = 1)
colnames(y) <- "y"
x.test <- x[1:10, ]
y.test <- y[1:10]
x <- x[-c(1:10), ]
y <- y[-c(1:10)]

mod <- gmdh.combi.twice(X = x, y = y, criteria = "PRESS")
pred <- predict(mod, x.test)
summary(sqrt((pred - y.test)^2))
```

Description

Build a regression model performing GMDH GIA (Generalized Iterative Algorithm) with Active Neurons (Combinatorial algorithm).

For more information, please read the package’s vignette.
Usage

gmdh.gia(X, y, prune = 40, criteria = c("PRESS", "test", "ICOMP"),
  x.test = NULL, y.test = NULL)

Arguments

x matrix with N>3 columns and M rows, containing independent variables in the model.
The data must not contain NAs

y vector or matrix containing dependent variable in the model.
The data must not contain NAs

prune an integer whose recommended minimum value is the number of initial regressors.
The maximum value will depend on the available RAM. It is recommended to work with the maximum value, but it can be computationally very expensive.
Prune is the selected number of neurons from layer i to layer i+1. The resulting layer i+1 has prune(prune-1)/2 neurons; for example with prune=150, the resulting neurons will be 11.175

criteria GMDH external criteria. Values:

  • PRESS: predicted residual error sum of squares.
  • test: use x.test and y.test to estimate RMSE (root mean square errors).
  • ICOMP: Index of Informational Complexity. Like PRESS, it is computed without recalculating of system.

x.test matrix with a sample randomly drawn from the initial data.
It is used when criteria = test.
This sample should not be included in X.

y.test vector or matrix with y values correspond with x.test values.

Value

An object of class gia.

References


Examples

```r
set.seed(123)
x <- matrix(data = c(rnorm(1000)), ncol = 5, nrow = 200)
colnames(x) <- c("a", "b", "c", "d", "e")
y <- matrix(data = c(10 + x[, "a"] * x[, "e"]^3), ncol = 1)
colnames(y) <- "y"
x.test <- x[1:10, ]
y.test <- y[1:10]
x <- x[-c(1:10), ]
y <- y[-c(1:10)]

mod <- gmdh.gia(X = x, y = y, prune = 5, criteria = "PRESS")
pred <- predict(mod, x.test)
summary(sqrt((pred - y.test)^2))
```

---

**gmdh.mia**

**GMDH MIA**

**Description**

Build a regression model performing GMDH MIA (Multilayered Iterative Algorithm).

For more information, please read the package’s vignette.

**Usage**

```r
gmdh.mia(X, y, prune = 150, criteria = c("PRESS", "test", "ICOMP"),
          x.test = NULL, y.test = NULL)
```

**Arguments**

- `X` matrix with N>3 columns and M rows, containing independent variables in the model.
The data must not contain NAs
- `y` vector or matrix containing dependent variable in the model.
The data must not contain NAs
- `prune` an integer whose recommended minimum value is the number of initial regressors.
The maximum value will depend on the available RAM. It is recommended to work with the maximum value, but it can be computationally very expensive.
Prune is the selected number of neurons from layer i to layer i+1. The resulting layer i+1 has prune(prune-1)/2 neurons; for example with prune=150, the resulting neurons will be 11.175

criteria GMDH external criteria. Values:

- PRESS: Predicted Residual Error Sum of Squares. It take into account all information in data sample and it is computed without recalculating of system for each test point.
- test: use x.test and y.test to estimate RMSE (Root Mean Square Errors).
- ICOMP: Index of Informational Complexity. Like PRESS, it is computed without recalculating of system.

x.test matrix with a sample randomly drawn from the initial data. It is used when criteria = test. This sample should not be included in X.

y.test vector or matrix with y values correspond with x.test values.

Value An object of class mia.

References


Examples

```r
set.seed(123)
x <- matrix(data = c(rnorm(1000)), ncol = 5, nrow = 200)
colnames(x) <- c("a", "b", "c", "d", "e")
y <- matrix(data = c(10 + x[, "a"] * x[, "e"]^3), ncol = 1)
colnames(y) <- "y"
x.test <- x[1:10, ]
y.test <- y[1:10]
x <- x[-c(1:10), ]
```
predict.combi

```r
y <- y[-c(1:10)]
mod <- gmdh.mia(X = x, y = y, prune = 5, criteria = "PRESS")
pred <- predict(mod, x.test)
summary(sqrt((pred - y.test)^2))
```

**predict.combi**  
**Predict GMDH Combinatorial**

**Description**

Calculates GMDH Combinatorial model predictions for new data.

**Usage**

```r
# S3 method for class 'combi'
predict(object, newdata, ...)
```

**Arguments**

- **object**: an object of class 'combi'
- **newdata**: matrix containing dependent variables in the model, which the predictions are calculated.
- **...**: other undocumented arguments

**Value**

A matrix with predictions.

**Examples**

```r
set.seed(123)
x <- matrix(data = c(rnorm(1050)), ncol = 3, nrow = 350)
colnames(x) <- c("a", "b", "c")
y <- matrix(data = c(10 + x[, "a"] + x[, "b"]^2 + x[, "c"]^3), ncol = 1)
colnames(y) <- "y"
x.test <- x[1:10, ]
y.test <- y[1:10]
x <- x[-c(1:10), ]
y <- y[-c(1:10)]
mod <- gmdh.combi(X = x, y = y, criteria = "PRESS")
pred <- predict(mod, x.test)
summary(sqrt((pred - y.test)^2))
```
**predict.combitwice**

*Predict GMDH Twice-Multilayered Combinatorial*

**Description**

Calculates GMDH Twice-Multilayered Combinatorial model predictions for new data.

**Usage**

```r
## S3 method for class 'combitwice'
predict(object, newdata, ...)
```

**Arguments**

- `object`: an object of class 'combitwice'
- `newdata`: matrix containing dependent variables in the model, with the predictions are calculated.
- `...`: other undocumented arguments

**Value**

A matrix with predictions.

**Examples**

```r
set.seed(123)
x <- matrix(data = c(rnorm(1050)), ncol = 3, nrow = 350)
colnames(x) <- c("a", "b", "c")
y <- matrix(data = c(10 + x[, "a"] + x[, "b"]^2 + x[, "c"]^3), ncol = 1)
colnames(y) <- "y"
x.test <- x[1:10, ]
y.test <- y[1:10]
x <- x[-c(1:10), ]
y <- y[-c(1:10)]
mod <- gmdh.combi.twice(X = x, y = y, criteria = "PRESS")
pred <- predict(mod, x.test)
summary(sqrt((pred - y.test)^2))
```
predict.gia  

Predict GMDH GIA object

Description

Calculates GMDH GIA Twice model predictions for new data.

Usage

```r
## S3 method for class 'gia'
predict(object, newdata, ...)
```

Arguments

- `object`: an object of class `giatwice`
- `newdata`: matrix containing dependent variables in the model, with the predictions are calculated.
- `...`: other undocumented arguments

Value

A matrix with predictions.

Examples

```r
set.seed(123)
x <- matrix(data = c(rnorm(1000)), ncol = 5, nrow = 200)
colnames(x) <- c("a", "b", "c", "d", "e")
y <- matrix(data = c(10 + x[, "a"] * x[, "e"]^3), ncol = 1)
colnames(y) <- "y"
x.test <- x[1:10, ]
y.test <- y[1:10]
x <- x[-c(1:10), ]
y <- y[-c(1:10)]

mod <- gmdh.gia(X = x, y = y, prune = 5, criteria = "PRESS")
pred <- predict(mod, x.test)
summary(sqrt((pred - y.test)^2))
```
predict.mia

Predict GMDH MIA object

Description
Calculates GMDH MIA model predictions for new data.

Usage
```r
## S3 method for class 'mia'
predict(object, newdata, ...)
```

Arguments
- `object`: an object of class 'mia'
- `newdata`: matrix containing dependent variables in the model, with which the predictions are calculated.
- `...`: other undocumented arguments

Value
A matrix with predictions.

Examples
```r
set.seed(123)
x <- matrix(data = c(rnorm(1000)), ncol = 5, nrow = 200)
colnames(x) <- c("a", "b", "c", "d", "e")
y <- matrix(data = c(10 + x[, "a"] * x[, "e"]^3), ncol = 1)
colnames(y) <- "y"
x.test <- x[1:10, ]
y.test <- y[1:10]
x <- x[-c(1:10), ]
y <- y[-c(1:10)]

mod <- gmdh.mia(X = x, y = y, prune = 5, criteria = "PRESS")
pred <- predict(mod, x.test)
summary(sqrt((pred - y.test)^2))
```